

### **REMARKS**

Claims 1-14 and 16-20 are pending in this application. Claims 1-14 and 16-20 are rejected. The rejections are respectfully traversed.

#### **CLAIM REJECTIONS – 35 U.S.C. § 102**

Claims 1-5, and 7-9 are rejected under 35 U.S.C. § 102(b) as being anticipated by Dinallo, US patent no. 5,727,212.

Claims 14 and 16-20 are rejected under 35 U.S.C. § 102(e) as being anticipated by Edmunds, US Pub no. 2003/0231329.

#### **Claims 1-5 and 7-9 Define over the Cited Art:**

Dinallo fails to disclose, either expressly or inherently, each and every feature of the claimed embodiments. Consider claim 1, which recites in part:

***upon receipt of a request for a feature, calling the generic routines as a function of the feature;  
identifying a peripheral device capable of performing a specific feature corresponding to the feature requested;  
executing a native driver of the identified peripheral device***

Dinallo fails to disclose, either expressly or inherently, calling the generic routines as a function of the feature, identifying a peripheral device capable of performing a specific feature corresponding to the feature requested and executing a native driver of the identified peripheral device. The Office Action relies upon the abstract and column 5, lines 2-25, to disclose the claimed features. Specifically, the Office Action relies upon the abstract of Dinallo to disclose the claimed "identifying a peripheral device capable of performing a specific feature corresponding to the feature requested." MPEP §706.02(II) recites in part:

***Citation of and reliance upon an abstract*** without citation of and reliance upon the underlying scientific document ***is generally inappropriate*** where both the abstract and the underlying document are prior art. See *Ex parte Jones*, 62 USPQ2d 1206, 1208 (Bd. Pat. App. & Inter. 2001) (unpublished). ... When both the abstract and the underlying document qualify as prior art, the underlying document should normally be used to support a rejection. ***In limited circumstances, it may be appropriate for the examiner to make a rejection in a non-final Office action*** based in whole or in part on the abstract only without relying on the full text document.

Accordingly, because the Office Action is final, the Office Action's reliance on the abstract of Dinallo is inappropriate. Applicants respectfully request withdrawal of the finality of the Office

Action and issuance of a new Office Action which does not inappropriately rely upon an abstract to form a rejection.

Dinallo discusses a DDTransport class in an object orientated system for communicating with a device. (See, Dinallo at column 5, lines 6-9). Dinallo discusses creating an object in an object orientated system (steps 504-506), establishing driver communication with a device through the DDTransport class (step 508) and then requesting that the device perform a command with respect to the created object (step 510). (See, Dinallo at column 5, lines 26-40 and FIG. 5). In contrast, the claimed embodiment provides for, upon receipt of a request for a feature, calling the generic routines as a function of the feature, identifying a peripheral device capable of performing a specific feature corresponding to the feature requested and then executing a native driver of the identified peripheral device. Dinallo fails to disclose, either expressly or inherently, the claimed features because Dinallo's system establishes communication with a device before requesting the device perform a command. That is, Dinallo fails to disclose **identifying a peripheral device capable of performing a specific feature corresponding to the feature requested** and executing a native driver of the identified peripheral device because Dinallo's system establishes communication with a device before sending the device a command, and, thus, there would be no need for Dinallo's system to identify a device capable of performing a feature based upon a requested feature because Dinallo's system is already in communication with the device before a command is issued. One benefit of the claimed embodiment, for example, is that a generic command to print, scan or fax a document, for example, can be requested and the claimed method can identify peripherals capable of performing the command before executing a driver for the identified peripheral.

For the foregoing reasons, withdrawal of the rejection of claim 1 is respectfully requested. Claims 2-5 and 7-9 refer to independent claim 1, and therefore patentably distinguish over the cited art.

**Claims 14 and 16-20 Define over the Cited Art:**

Edmunds fails to disclose, either expressly or inherently, each and every feature of the claimed embodiments. Consider claim 14, which recites in part:

a mobile computer configured to provide **an interface** used by an application to access the at least one peripheral device, to use the interface **to call a plurality of generic routines as a function of a request for a feature**, the generic routines **to cause the native driver, installed on the mobile computer, to**

***execute and control the peripheral device and perform a specific feature corresponding to the feature requested, the interface being independent of device-specific features of the at least one peripheral device.***

The Office Action, asserts the "generic driver interface" of Edmonds discloses the claimed "interface," citing paragraphs [0007-0011] of Edmonds. However, Applicants respectfully submit that the generic driver interface of Edmonds cannot disclose, either expressly or inherently, the claimed "interface ... to call a plurality of generic routines ... the generic routines to cause the native driver ... to execute and control the peripheral device," as recited in claim 1, because Edmonds generic driver interface uses a generic driver to generate print jobs solely on USB printers. (See, Edmonds at paragraph 23, lines 1-6). That is, Edmonds' generic driver interface does not use the drivers native to the peripheral device.

The Office Action, asserts that Edmonds at paragraph [0007] discloses the claimed "as a function of a request for a feature, the generic routines to cause the native driver." However, paragraph [0007] discusses a Beacon printer, which Edmonds describes as a network printer interface, not a USB printer interface. The generic driver interface of Edmonds could not field requests to the beacon printer, because the beacon printer, not being connected to a USB port, would not respond to generic USB printer command on a USB port.

Furthermore, Edmonds' beacon interface is located at the beacon printer itself. See, for example, paragraph [0018] of Edmonds which discusses that the user must physically walk to the desired beacon printer and select the printer through some interface resident on the printer. Accordingly, Applicants respectfully submit that Edmonds' beacon printer interface fails to disclose the claimed "interface ... to call a plurality of generic routines ... the generic routines to cause the native driver ... to execute and control the peripheral device," because the beacon interface only allows access to the single device which the user has chosen to walk to.

Further still, Edmonds beacon driver "must be capable of translating a print job from an application ***into the appropriate page description language (PDL) for any supported printer.***" (See, Edmonds at paragraph 15, lines 8-10). Accordingly, Applicants respectfully submit that Edmonds beacon interface fails to disclose, either expressly or implicitly, the claimed "to use the interface ***to call a plurality of generic routines as a function of a request for a feature,***" because the beacon interface uses a PDL specific to the selected printer. That is, there is no evidence that Edmonds beacon printer interface calls generic

routines, instead the beacon driver must call specific routines to translate a print job into the appropriate PDL.

Furthermore, Applicants respectfully submit that neither the beacon printer interface nor the generic USB printer interface of Edmonds disclose the claimed "the interface being independent of device-specific features of the at least one peripheral device," because the beacon printer interface and the generic USB printer interface are both tied to the specific features of a printer. Furthermore, the Office Action fails to cite any art against this feature of the claimed embodiment. Accordingly, Applicants respectfully request withdrawal of the finality of the outstanding Office Action and issuance of a new non-final Office Action, if necessary, addressing each and every feature of the claimed embodiments.

Accordingly, Edmonds' fails to disclose, either expressly or inherently, the claimed "interface ... to call a plurality of generic routines ... the generic routines to cause the native driver ... to execute and control the peripheral device driver ... the interface being independent of device-specific features of the at least one peripheral device," because Edmonds' generic USB printer driver interface uses a generic USB printer driver, not the driver native to the peripheral device and Edmonds' beacon printer driver interface only allows access to a single printer and does not call generic routines.

For the foregoing, Applicants respectfully request withdrawal of the rejection to claim 14. Dependent claims 16-20 refer to independent claims 14 and, therefore, are allowable over this art.

#### **CLAIM REJECTIONS – 35 U.S.C. § 103**

Claims 6, 11, 12, and 13 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Dinallo, US patent no. 5,727,212 in view of Edmonds, US pub. no. 2003/0231329.

Claim 10 is rejected under 35 U.S.C. § 103(a) as being unpatentable over Dinallo, US patent no. 5,727,212 in view of Dorris, US patent no. 5,867,710.

Dinallo and Edmonds, even when considered in combination, fail to disclose, either expressly or implicitly, each and every feature of the claimed embodiments.

The Office Action, in rejecting claim 12 at page 8 of the Office Action, asserts: "Edmunds teaches ... if the request is a request to connect a computer to the requested peripheral device, ***cause the device to execute*** ..." However, independent claim 12, recites in part:

if the request is a request to connect a computer to the requested peripheral device and the requested peripheral device is accessible,

***instantiating the connection class to create an object specific to the requested peripheral device,***  
***using the instantiated object to cause a native driver of the requested peripheral device to execute, and***  
***connecting, through the native driver, the computer to the requested peripheral device,***

The Office Action fails to address these features. Accordingly, withdrawal of the finality of the present Office Action and issuance of a new, non-final Office Action addressing each and every feature of the claimed embodiments is respectfully requested.

Further, the Office Action, at page 8, appears to rely upon Edmunds' discussion of a beacon printer to disclose the features of the claimed embodiments. As discussed above, Edmunds' beacon interface is located at the beacon printer itself. See, for example, paragraph [0018] of Edmunds which discusses that the user must physically walk to the desired beacon printer and select the printer through some interface resident on the printer. However, there is no evidence that Edmunds' beacon interface instantiates a connection class to create an object specific to the requested peripheral device, uses the instantiated object to cause a native driver of the requested peripheral device to execute, and connects, through the native driver, the computer to the requested peripheral device, because Edmunds merely discusses that after the user selects a printer, "the user's print job is directed to that printer." (See, Edmunds at paragraph [0009], lines 10-13).

Furthermore, Dinallo fails to disclose the claimed "connecting, ***through the native driver***, the computer to the requested peripheral device." Dinallo discusses using a DDTransport class and a DDInterface to connect device drivers to an object orientated subsystem because the device drivers discussed in Dinallo were incapable of connecting to the object orientated subsystem without rewriting the computers operating system and its device drivers to the standards of object orientated programming. (See, for example, Dinallo at FIGS. 2 and 6 and at column 1, lines 39-41 and column 5, lines 3-9).

Accordingly, Applicants respectfully submit that a *prima facie* case of obviousness cannot be based upon Dinallo and Edmunds, because there is no evidence that one of ordinary skill in the art would combine Edmunds' beacon printer interface with Dinallo's object orientated subsystem/device driver interface and modify the combination to include "instantiating the connection class to create an object specific to the requested peripheral device, using the instantiated object to cause a native driver of the requested peripheral device to execute, and

connecting, through the native driver, the computer to the requested peripheral device," as recited in claim 12.

For the foregoing reasons, withdrawal of the rejection of claim 12 is respectfully requested. Dependent claim 13 refers to claim 12, and therefore patentably distinguishes over the cited art.

Further, dependent claims 6, 10 and 12 refer to independent claim 1, and therefore patentably distinguish over the cited art.

### **CONCLUSION**

For all the above reasons, the applicants respectfully submit that this application is in condition for allowance. A Notice of Allowance is earnestly solicited.

The Examiner is invited to contact the undersigned at (202) 220-4200 to discuss any matter concerning this application.

The Office is authorized to charge any underpayment or credit any overpayment to Kenyon & Kenyon LLP's Deposit Account No. 11-0600.

Respectfully submitted,

KENYON & KENYON LLP

Date: December 8, 2008

/Matthew H. Polson/

Matthew H. Polson  
(Registration No. 58,841)

Kenyon & Kenyon LLP  
1500 K Street, NW  
Washington, DC 20005-1257  
Telephone: (202) 220-4200  
Facsimile: (202) 220-4201